In the Claims:

1-27 (cancelled)

28. (new) In an object-oriented computer system, a method of generating software components, the method comprising the steps of:

Loading a structure of an object to a memory means;

Loading the file structure data by a file component loader;

Interfacing between the structure of an object and the file structure data by an interface object;

interfacing between the structure of an object and the file structure data by an interface object using the methods InitComplexRepository(), GetComplexData() and PutComplexData().

having said structure contains a plurality of pointers to complex functionalities and complex data storage buffers;

Loading the file structure data by a file component loader which includes complex functionality and a knowledge base; and

having said structure contains a plurality of pointers to complex functionalities and complex data storage buffers and said complex data storage buffers includes metadata models.

29. (new) The method of claim 28 further comprises:

said structure contains a plurality of pointers to complex functionalities and complex data storage buffers and said complex data storage buffers includes pattern recognition components.

30. (new) In an object-oriented computer system, a device to generate software components comprising of:

a load means for loading a structure of an object in a memory means;

a load means for loading the file structure data by a file component loader;

an interface means for interfacing between the structure of an object and the file structure data by an interface object;

the interface means between the structure of an object and the file structure data by an interface object using the methods InitComplexRepository(), GetComplexData() and PutComplexData();

and said structure contains a plurality of pointers to complex functionalities and complex data storage buffers and said complex data storage buffers includes metadata models.

31. (new) The device of claim 30 in which:

said structure contains a plurality of pointers to complex functionalities and complex data storage buffers and said complex data storage buffers includes pattern recognition components.

32. (new) A computer program product for generating software components, the computer program product comprising a computer usable medium having computer readable program code thereon, including: program code for loading a structure of an object in a memory means;

program code for loading the file structure data by a file component loader;

program code for Interfacing between the structure of an object and the file structure data by an interface object;

said base component has interfaces and the program code for: interfacing between the structure of an object and the file structure data by an interface object using the methods InitComplexRepository(), GetComplexData() and PutComplexData();

said base component has interfaces and the program code for: having the structure containing a plurality of pointers to complex functionalities and complex data storage buffers; and

said structure to contains plurality of pointers to complex functionalities and complex data storage buffers and said complex data storage buffers includes pattern recognition components.

33. (new) The computer program product of claim 32 wherein the base component has interfaces and the program code for:

Allocating an object in a memory means;

Copying the structure to a memory means;

Placing the records of complex data, complex functionality and knowledge base in the Complex Data Storage Buffer;

Creating complex data at instantiation moment;

Modifying complex data;

Deciding whether to write complex data, complex functionality and knowledge base to a memory means:

Exposing the structures and functionality through an interface that performs all specific task of the object itself and administrative tasks related to the structure; and

Using a object in which all structures can be used like memory structures and all memory allocations and swapping will be managed by the object.